DOCUMENT RESUME

ED 239 609 IR 020 060

AUTHOR	Stocker, Charlene
TITLE	Beginning a Study Investigating Computer Programming.
INSTITUTION	Area Education Agency 7, Cedar Falls, IA.
PUB DATE	[80]
NOTE	103p.
PUB TYPE	Guides - Classroom Use - Materials (For Learner) (051) -- Guides - Classroom Use - Guides (For Teachers) (052)
EDRS PRICE	MF01/PC05 Plus Postage.
DESCRIPTORS	*Computer Literacy; Elementary Education; Elementary School Mathematics; *Independent Study; Instructional Materials; Learning Activities; *Microcomputers; Programed Instructional Materials; *Programing; *Programing Languages; Sequential Approach
IDENTIFIERS	Apple (Computer); *BASIC Programing Language; PF Project

ABSTRACT

Intended for children from kindergarten through the sixth grade, this booklet teaches about the computer and provides instruction in computer programming in the BASIC language. Thirteen sequential lessons are provided to encourage independent study. Each lesson consists of a set of four pages and explains a BASIC command or statement, presents several activities, and includes a brief test. The booklet is written for use with the Apple microcomputer; however, instructions are given for adapting the material to the TRS-80 and the TI-99 microcomputers. Following the first page, which provides an overview and instructions for the teacher, the student materials are presented. The first lesson includes an introduction, a description of the parts of the computer, and an explanation of BASIC commands. The commands and topics which are covered in the remaining 12 lessons are: "print"; "goto"; "input"; "if then"; writing mathematics programs; "for next"; use of "greater than" and "less than" symbols; random numbers; variables; use of BASIC in mathematics; "gosub"; "read"; and "data." A test answer key and a student contract are included. This document is part of a collection of materials from the Iowa Area Education Agency 7 Teacher Center project. (DC)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED239609

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Ronald Dickinson

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

Beginning
A
Study
Investigating
Computer Programming

IR020060

Developed By:

Charlene Stocker
Classroom Teacher
Oelwein Community Schools
Oelwein, Iowa

Sponsored By:

AEA-7 Teacher Center
Nellie Garvey School
2201 E. Mitchell Ave.
Waterloo, Iowa 50702

Cedar Falls, Iowa

Document title: Beginning A Study Investigating Computer Programming.

Author(s): Charlene Stocker

Developed with funds provided by a federal Teacher Center grant contract under the provisions of the Higher Education Act of 1965.

AEA 7 Teacher Center
Waterloo, Iowa
Sept. 15, 1978 to Aug. 15, 1983

Teacher Center Policy Board Chair - Doris Mauer
Teacher Center Director - Diane Gibson, Ed. D.

Ronald Dickinson, Ed. D.- Chief Administrator, AEA 7

TO THE TEACHER

This booklet has been prepared for elementary grade children who are ready and anxious to learn more about their computer and how it operates. The series of 13 lessons is designed to encourage independent study by children of any grade level. Children in kindergarten and first grade have used and enjoyed the first few lessons. So have children in sixth. As the children progress through the lessons, the material becomes increasingly difficult. No matter what grade the child is in, it is important that he begins at the beginning.

As the children use the material they will be exposed to BASIC commands, statements and operations of the computer. They will learn about:

RUN	PRINT	THEN	GOSUB	READ	* ↑
LIST	GOTO	FOR	RETURN	DATA	/
NEW	INPUT	NEXT	CLEAR	$	=
HOME	TAB	REM	END	=	<
CONT	IF	STEP	RND	-	>

Each lesson explains the BASIC command or statement, presents several activities for the children to practice and includes a brief test. The lessons have been grouped in sets of four pages. This allows you to attach each lesson to a file folder, laminate it, and place the lessons near the computer. The children seem to enjoy the convenience of handling one file folder as opposed to using a text book.

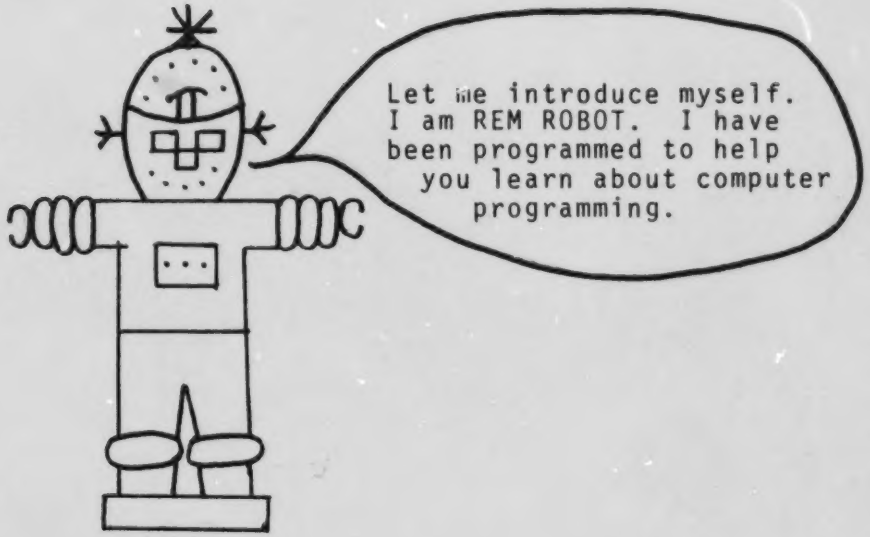
There is no material covering graphics in these lessons. Because of the variation from one brand of computer to another, any mention of graphics would have limited this booklet's use to one computer.

The booklet is written for use with the Apple. There are a few minor changes for other brands of computers. These changes are:

TRS-80:
1. change all references from HOME to CLS
2. use BREAK instead of Cntl-C
3. use the ENTER key instead of RETURN

TI-99
1. time loops are written FOR DELAY= etc. NEXT DELAY
2. use function 4 to break into a program
3. spaces are significant in a program
4. use CALL CLEAR instead of HOME
5. check the manual for using the string functions of the TI-99
6. use the ENTER key instead of RETURN

This booklet is a beginning. A beginning that children can understand, enjoy, and leave them asking for more.



INTRODUCTION

Let me introduce myself.

I am REM ROBOT.

I have been programmed to help you learn about computer programming.

Some of you may be 5 years old. Some may be 10. Some may be 90. Any age is fine. You're never too young or too old to learn about computer programming.

I have written these lessons so you will be able to sit down - by yourself - at a computer and experiment.

The first thing you will be doing is learning to read the language the computer uses. BASIC.

Then you will learn how to put those BASIC statements together to make a computer program that works.

It's really quite simple. You will find many activities for you to try. They will be enclosed in boxes like this: Be sure and try all of them. Each program you type (and each mistake you make) will make the next lesson easier.

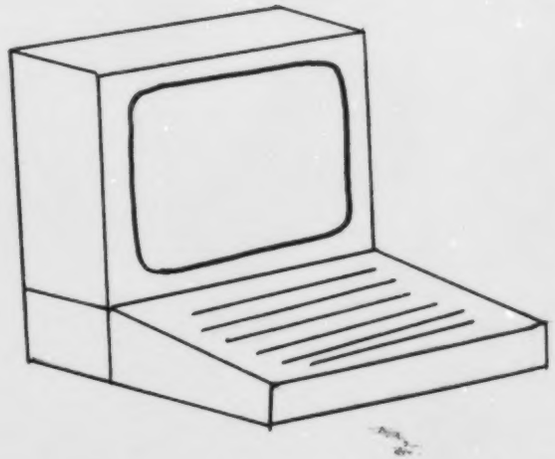
When you finish this book, you will be able to write programs that will help you, programs that your friends can use, programs that your teachers can use, and you will be able to understand programs written by others. But best of all, you will have learned how to communicate with a computer and how to make it do what you tell it to do.

SOME VERY IMPORTANT COMPUTER INFORMATION

As you look at your computer you see a keyboard. This works like a typewriter. You send information to the computer through this keyboard.

The TV screen in front of you is where everything is displayed; the programs you type and the programs you run. It is connected to the computer. You can turn the TV off and on again without losing your program.

Our computer has two kinds of memory. The commands (or instructions) the computer needs in order to work are stored in the permanent memory. It is called the ROM memory. ROM stands for READ ONLY MEMORY. The information in the ROM is put in when the computer is made and it stays there whether the computer is turned on or off.



The RAM memory (RANDOM ACCESS MEMORY) is for us to use. This is like blank pages in a book. When we write programs, we use the RAM memory. Since our microcomputers are so small, we do not SAVE or store our programs inside the computer. We use cassette tapes or disks. Programs in the RAM memory are erased when the computer is turned off. Unless we have SAVED them on tapes or disks we cannot get them back without re-typing them.

The Central Processing Unit (CPU) cannot be seen. It is inside the computer. It translates the information typed by you into signals the computer can use and understand. And it translates the action of the computer into words we can use and understand. We don't have to know how it works. Only that it does!

The BASIC language is really just simplified English. BASIC stands for Beginners All-purpose Symbolic Instruction Code.

To give you an idea of just how simple it is, let's use the command RUN for an example.

RUN really says to the computer - find a program that is in your memory - go to the beginning of that program - read and follow the directions given to you.

Do you see why just typing RUN is a lot easier than typing all those words!

BASIC COMMANDS

IT IS VERY IMPORTANT THAT YOU SPELL THESE COMMANDS CORRECTLY WHEN YOU TYPE THEM.
IF YOU DON'T, THE COMPUTER WILL PRINT SOME KIND OF ERROR MESSAGE AND WILL NOT DO ANYTHING UNTIL YOU CORRECT THE SPELLING!
THE COMPUTER CARRIES OUT YOUR COMMANDS IMMEDIATELY!

RUN Typing this command instructs the computer to begin reading any program that is in it's memory.

RETURN This command is given by pressing the RETURN key on the keyboard. It is a signal that tells the computer you have finished typing and are ready for the computer to work. Some computers use the command ENTER and have an ENTER key.

Cntl C Pressing the control key and typing the letter C will allow you to stop a program that is running. Some computers have a key labeled BREAK for this purpose.

CONT You may start a program that has been stopped by typing RUN to go back to the beginning of the program or by typing CONT which will tell the computer to continue from where it left off.

LIST Typing this command lets you see the list of instructions you have typed - in other words - your program.

NEW Completely erases everything in the RAM memory.

HOME Erases the TV screen but does not erase the information in the RAM memory. This command varies from computer to computer. Check your manual. It may be CLS or CALL CLEAR. This command is also a program statement and can be used when writing your programs.

PRINT Tells the computer to PRINT everything that follows on that line. This command is also a program statement. Check your computer. Some let you type a ? instead of typing the word PRINT.

SAVE Tells the computer to save your program on cassettes or disks. Check your computer manual for the exact procedure.

Now that you have done all that reading, let's get our hands on that computer! Turn it on. Press the RESET button if you have one and type NEW to erase anything that is in the computer's RAM memory.

1. Start typing. You won't hurt anything. Find out what all of those keys do.
2. What happens when you hold the shift key down and type?
3. What does the backwards arrow do? Does your computer have a repeat key? What does it do?
4. Be sure to notice that there is one 0 for numbers and another one for letters. The one with the 1 is for numbers. It is very important that you do not type a number 0 for a letter O. The computer won't work if you mix them up.
5. Type your name. Type my name.
6. What does the space bar do? Can you fill the screen with letters and numbers and symbols?
7. When you are sure you have tried out all the keys, then go on to the next lesson. You'll learn your first BASIC program statement, PRINT, and what PRINT will do for you.

TEST YOURSELF

1. A machine that can store and use information, but only does what it is told to do is a:
   1. TV
   2. microcomputer
   3. lawnmower

2. The TV screen is used to:
   1. display computer output
   2. watch cartoons
   3. watch you do your work

3. The keyboard is used to:
   1. write letters home to mother
   2. send input to the computer
   3. send messages to Mars

4. The "brain" of the computer that translates the information we give into signals the computer can use is:
   1. the principal
   2. computer operator
   3. Central Processing Unit (CPU)

5. The portion of the microcomputer that stores information is:
   1. Tupperware
   2. memory
   3. CPU

6. Computer programs can be stored on:
   1. scotch tape
   2. cassette tapes or disks
   3. candy dishes

7. A simple type of language we use to tell computers what to do is:
   1. babytalk
   2. Brazilian
   3. BASIC

PRINT

Your computer has been programmed to read the word PRINT. When PRINT is typed, the computer will print whatever you tell it to PRINT.

If you want the computer to print words, put the words inside quotation marks like this: " "

```
PRINT "READ"
PRINT "ON"
```

Activity #1

Type this:

```
PRINT "HELLO"

PRINT "(type your name)"
```

Be sure to press the RETURN or ENTER key when you finish typing.

The computer printed the word HELLO and it printed your name.

It's pretty simple to make that computer work, isn't it?

Activity #2

Type this:

```
PRINT "HOLLO"
```

This time make a spelling mistake.

The computer printed the mistake! It will print everything you enclose in quotation marks exactly as you type it. If you type a mistake, it will print a mistake.

You can correct those mistakes by using the backwards arrow to "eat up" the wrong letters. Then re-type the correct letters.

If you have already pressed RETURN, just re-type the line.

Activity #3

Tell the computer to print numbers.

```
PRINT "3  4  5"

Print  3  4  5
```

You do not need to use quotation marks around numbers.

Sometimes you will want quotation marks around numbers, sometimes you won't. We'll talk about this later.

You have been using PRINT as a command.
You typed PRINT - and the computer PRINTED something immediately.

PUT A LITTLE PRINT IN YOUR PROGRAM

This time we'll write a computer program. A computer program is a set of instructions the computer must follow. Computer programs have line numbers. The line number is your way of saying to the computer, "Computer, pay attention. I'm writing a program." Do not do what I tell you to until I type RUN.

Activity #4

(Be sure to press RETURN after you have finished typing each line.)

Type this:

```
10 PRINT "ROSES ARE RED."
20 PRINT "VIOLETS ARE BLUE."
30 PRINT "I'M WRITING A PROGRAM."
40 PRINT "WHOOPEE AND YA HOO!"
```

In this program you are using PRINT as a program statement. The computer will not pr int the things you tell it to until you type RUN.

Do you see the list of instructions you typed? (your program) And the program the computer printed? (That's the one without the line numbers and the quotation marks.)

Activity #5

How to do it.

Add this line to your program and watch the neat thing that happens.

```
5 HOME
```

1. type LIST
2. type line 5 HOME
3. RUN your program

Look! The list of instructions is gone.

Type LIST and look at those numbers.
Line 5 is right up there on top.
The computer put it in the right order.
Wasn't that nice of it?

Activity #6

The computer will print blank spaces for you - if you tell the computer that is what you want it to do. I will show you what I mean. Type NEW to erase your old program. Then type this:

```
10 HOME
20 PRINT"ZZZZZ"
30 PRINT"      ZZZZZ"
40 PRINT"            ZZZZZ"
50 PRINT"WAKE UP!"
```

Your words don't have to begin at the left side of the TV screen. You can tell the computer to leave blank spaces and it will!

MORE PROGRAMS USING PRINT

Activity #7

Since we know our computer will print spaces, let's draw a rectangle. That should be easy to do.

```
10 HOME
20 PRINT "    ####################
30 PRINT "    #                  #
40 PRINT "    # (type your name) #
50 PRINT "    #                  #
60 PRINT "    ####################
```

My, your name looks nice there.

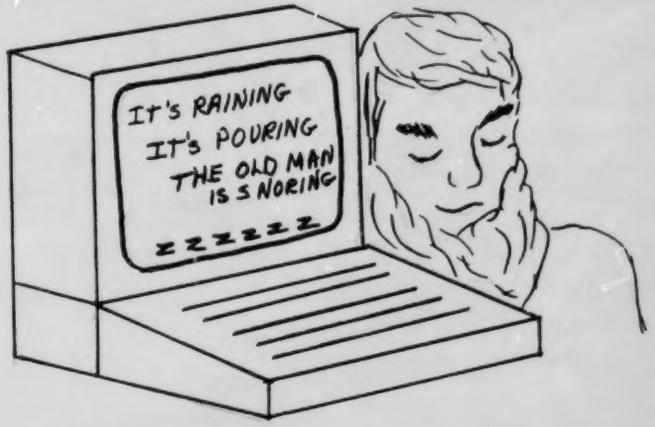
When you write long programs and save them for other kids to use, you will want to sign your name. This is one way that looks good.

If you change your mind and want to change a line, just type the same line number and type a new line. The old line will disappear forever!

Activity #8

You can make your computer skip lines when it prints things on the TV screen. Sometimes your programs are easier to read this way.

```
10 HOME
20 PRINT "IT'S RAINING"
30 PRINT
40 PRINT "     IT'S POURING"
50 PRINT
60 PRINT "          THE OLD MAN IS SNORING"
70 PRINT
80 PRINT " ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ"
```



Lines 30, 50, and 70 told the computer to print nothing at all. And that is exactly what it did!

You have learned about the BASIC statement PRINT and some of the things you can make the computer do when you use it. You are beginning to talk to your computer.

I hope you haven't forgotten the short cut way ot telling the computer to PRINT - that's right, use the?

TEST YOURSELF

1. A set of instructions the computer must follow is a:
   1. program
   2. menu
   3. dictionary

2. Commands are instructions the computer:
   1. carries out immediately
   2. follows in a program
   3. pays no attention to

3. Statements are instructions the computer:
   1. carries out immediately
   2. follows in a program
   3. pays no attention to

4. Statements must have:
   1. a period
   2. quotations
   3. a line number

5. The computer reads the lines of a program:
   1. in numerical order
   2. in the order you typed them
   3. with it's reading glasses on

6. Pressing the ENTER key instructs the computer to:
   1. act on whatever you have typed
   2. turn on
   3. read the program again

7. The PRINT statement instructs the computer to:
   1. print whatever you type
   2. make copies of your lesson
   3. wait while you get a pencil

8. You can change lines in your program by:
   1. asking your teacher
   2. re-typing a line using the same line number as the one you want to change
   3. you can't do it

9. Spelling mistakes in commands and statements:
   1. are OK because the computer will correct them
   2. will cause the computer to wait until you correct them
   3. will give your teacher a headache

10. Words inside quotation marks:
    1. must be spelled correctly or the computer won't work
    2. are called ribits
    3. are printed exactly as you spell them

11. The ? is a quick way of typing:
    1. a question
    2. PRINT
    3. nothing at all

12. `10 ?` This statement:
    1. Will make you scratch your head
    2. who knows?
    3. instruct the computer to leave a blank space in your program

GOTO

Go to bed. Go to Grandmas. Go to school. Go to sleep. That is what people have been telling you all your life. And that is what you can tell the computer. It has been programmed to understand what go to means - If you spell it GOTO.

We're going to make our computer programs go around in circles with GOTO.
Before you begin typing this lesson, be sure you know how to stop a computer program.
Is it Cntl C or BREAK or something else?

Once we send the program off on this kind of a GOTO circle, we won't be able to stop it any other way - unless we turn the computer off!

Activity #1

Type this:

```
 1  HOME
10  PRINT "(type your first name here)"
20  PRINT "      (type your last name here)"
30  GOTO 10
```

RUN this and watch your name fly by! Was that ever fast!
You told the computer to print your name and then to go back to the beginning and do it again. And again. And again.

Stop the program and I'll show you a way to slow it down a bit.

Type this:

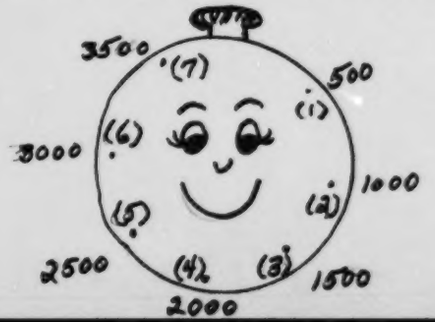
```
25 FOR T = 1 to 50: NEXT
```

RUN it again. That was a little better wasn't it?

The line you added (line 25) was a time delay loop. Don't wonder about how it works for now. That explanation will come later.

For now, just remember that the number 500 will cause the computer to hold your sentence on the screen for 1 second.

The number 1000 for 2 seconds
1500 for 3 seconds
2000 for 4 seconds
2500 for 5 seconds



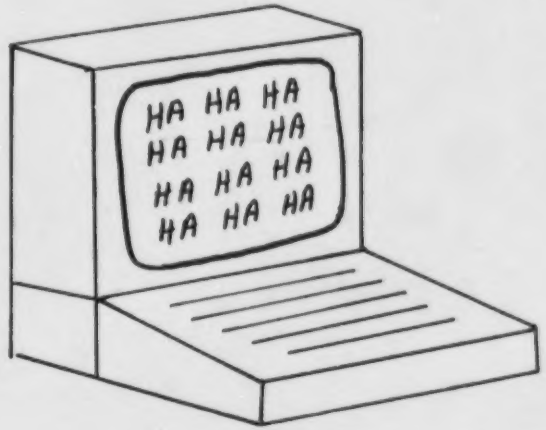
YOU CAN MAKE YOUR COMPUTER LAUGH (HA HA) 10
YOU CAN MAKE YOUR COMPUTER CRY (BOO HOO)
YOU CAN MAKE YOUR COMPUTER SING (TRA LA LA)
SO GIVE IT A TRY

Activity #2

We're going to fill the screen with laughter.
Type this:

```
 1  HOME
10  PRINT " HA HA "
20  GOTO 10
```

Be sure and pay attention to the spaces before and after the word HA HA.
Run your program.
That was neat but it didn't fill the screen did it?



This time, put a semicolon ; at the end of the line.
Type a new line 10 that looks like this:

```
10 PRINT " HA HA " ;
```

Now, try a comma

```
10 PRINT " HA HA " ,
```

Punctuation is important to the computer.

If all this is going by too fast, add a time delay loop.

```
15 FOR T = 1 TO 50: NEXT
```

Activity #3

See what kinds of tricks you can play with the two line program we just wrote.
Change line 10 so the computer will print something else. Make the computer cry. Make it sing.
Make it print your name. Make it print the names of your friends.

And be sure and tell it that it has done a good job. "GOOD JOB OLD BUDDY COMPUTER!"

COMPUTER CHEERLEADING

I suppose you've never heard of a computer cheerleader.
In this activity, you will use the time delay loops and GOTOs to make a cheer.

Activity # 4

Here's our first cheer:

```
10 PRINT " GO "
30 PRINT " TEAM "
50 PRINT " GO "
```

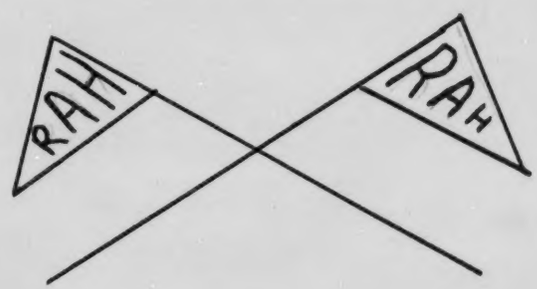
Those are the words. Now for the rhythm. (the time delay loops)

```
20 FOR T = 1 TO 250: NEXT
40 FOR T = 1 TO 250: NEXT
60 FOR T = 1 TO 500: NEXT
```
(This time delay is twice as long as the others.)

Now is you are a bit confused by the way we typed this program, type LIST and the computer will put everything in order. As you can see, each time delay loop comes after a word.

So far, so good. Add your GOTO statement:

```
70 GOTO 10
```



And a blank space between each cheer would be nice too.

```
65 PRINT
```

And of course, we should start off with a clear screen.

```
1 HOME
```

RUN this cheer and cheer with the computer. (Not too loud - unless you are writing this in a football stadium.)

Activity #5

This is a snap. Just change lines 10, 30, and 50 to make a new cheer. Here's a suggestion

WE'RE FROM IOWA WE'RE NUMBER ONE COMPUTERS ARE FUN GOODBY FOR NOW!

TEST YOURSELF

1. GOTO:
   1. sends the computer to another line of your program
   2. sends you to the classroom
   3. sends the computer to the end of your program

2. Pressing the BREAK key instructs the computer to:
   1. break
   2. stop the program immediately
   3. load another program

3. A time loop statement will:
   1. instruct the computer to hold your sentences on the screen
   2. let you plan a new program
   3. tell you what the correct time is

4. Spaces inside quotation marks:
   1. do not matter
   2. let you plan where you want words to appear on the screen
   3. break the computer

The INPUT Statement (Sub-title - The Question Maker)

Your computer has been programmed to read the statement INPUT.

You use INPUT when you want to ask a question and want someone to answer.

Activity #1

Type this:

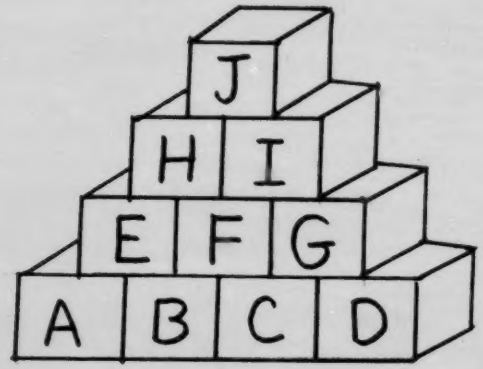
```
10 PRINT  "HOW OLD ARE YOU?"
20 INPUT A
30 PRINT "YOU LOOK GOOD FOR " A " YEARS OLD."
```

Now RUN it and see how this works.

I hope you paid attention to the letter A.
It is important. INPUT won't work without it.

The computer needs to remember the answer to the INPUT statement. So it has very conveniently set aside some storage boxes and labeled them with alphabet letters. (If you had 26 storage boxes, you'd want them labeled so you wouldn't forget which box you put something in!)

STORAGE BOXES



When you use INPUT, tell the computer in which storage box to put the answer. In our program, line 20 said put the answer in box A. Then in line 30 we told the computer to go to the A storage box and print whatever was there.

Simple isn't it?

What would have happened if we had left the A inside the quotation marks like this: "YOU LOOK GOOD FOR A YEARS OLD"

That's right! The computer would have printed the letter A. That would have made us look pretty silly.

The computer has another set of storage boxes for words.
The plain old A,B,C, etc. boxes are strictly for numbers.
To get into the storage boxes for words, you have to type the symbol $.
$ stands for String - and a string of letters is computer talk for a word or words.

Ativity #2

You have used computer programs that ask you to type your name and then the computer prints you name on the screen.
This is how to do it:

```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 ?"HELLO  " N$  "  GLAD TO MEET YOU."
```

Run this one to see if it works.
We could have used any alphabet letter.
I picked N because name begins with N and it was easy for me to remember.

Run this program again and type a different name. Maybe Snoopy. Or Mortimer.
The computer doesn't care what name you type.

I hope you noticed that when the computer read the INPUT statement, it put a ? on the screen and nothing else happened.
The computer will sit there for hours until someone types in something to answer the INPUT statement.

Did you also notice that you had to leave a space at the beginning or end of your PRINT statements when the storage box letter was also in the sentence.
That's because the computer can't tell where one word ends and another begins.
You have to tell it to leave spaces or it won't.

Activity #3

Are you ready for a program using lots of storage boxes?

```
 1  HOME
10  PRINT "WHAT IS YOUR NAME?"
20  INPUT N$
30  PRINT N$ " , TELL ME WHERE YOU LIVE."
40  INPUT L$
50  PRINT L$ " IS A NICE PLACE TO LIVE, " N$
60  PRINT "WHAT IS YOUR FAVORITE FOOD, " N$
70  INPUT F$
80  PRINT N$ ", FROM " L$ " , LIKES TO EAT " F$ "."
90  PRINT "GOOD BY," N$ "IT WAS NICE TALKING TO YOU."
```

You are really reading BASIC now. Just look at all those strange words and letters.
They really aren't so strange now that you know what they can do.

Activity #3

I want to very quickly introduce you to TAB.
Typing TAB with a number after it is another way of telling the computer you don't want it to print things on the left side of the screen.
It's a quick way of telling the computer to PRINT spaces.

Add this line to the program in Activity #3:

```
40 PRINT TAB (10), "#######"
```

Now that you see how easy TAB works, you can use TAB to print words, letters, and symbols where you want them.

Activity #4

Now that you understand that, I'm going to show you a short cut.

INPUT can be used instead of PRINT (providing you are asking a question).

```
10 INPUT "WHAT IS YOUR NAME " ; N$
20 PRINT N$
```

Be sure to include that ; semicolon.
From your work with GOTO, you should know what the ; will do.
That's right. The question mark will be on the same line as the question.
Do you need to include a question mark at the end of your question?

Try it <u>with</u> and <u>without</u> so you will know the answer.

Now that you know a short cut, try a program of your own.
Think up some good questions that your teacher will enjoy answering.

See you next lesson for IF THEN.

IF you are ready, THEN we will go on!

TEST YOURSELF

1. The INPUT statement instructs the computer to:
   1. to put another line in your program
   2. put some lights in the program
   3. to wait for some input

2. INPUT A instructs the computer to:
   1. wait for a number
   2. wait for a word
   3. wait for lunch

3. INPUT A$ instructs the computer to:
   1. wait for a number
   2. wait for a word
   3. wait for recess

4. $ is computer language for
   1. string
   2. money
   3. a nap

5. You must allow for spaces between words in your program because:
   1. The computer is just a machine that only does what it is told.
   2. No, you don't have to. The computer will fix it.
   3. I said so.

6. The computer uses its storage boxes to:
   1. store it's warm clothes
   2. remember the input it was given
   3. play store with them

Lesson 4

IF THEN

IF you clean your room, THEN you can play.
IF you eat your vegetables, THEN you can have your dessert.

You certainly know the meaning of the words IF and THEN.
So does your computer.

When you use IF and THEN in your programs you will be able to write programs that give the person using your program a choice.

Type this program. I'll explain each line as we go along.

Activity #1

```
10 HOME
     starts you off with a clean screen

20 INPUT "DO YOU LIKE BANANAS?  ANSWER YES OR NO"; A$
     the input statement allows for a question to be asked
     the question is printed on the screen
     the person using the program is told how to answer the question
     the answer  they give is put in the A$ storage box

30 IF A$= "YES" THEN 60
     if the person using the program types the letters YES, then the
       computer is told to skip all lines until it reaches line 60

40 PRINT "TOO BAD"
     if the person using the program typed anything except YES, then
       the computer prints the message TOO BAD

50 END
     the end of the program if the answer to the question was anything
       except yes

60 PRINT " GOOD, CAN I TRADE THIS BANANA FOR YOUR APPLE?"
     the message that is printed if the person using your program
      answered YES
```

That sure took a lot more words to explain than to write in BASIC. Do you see why BASIC is called simplified ENGLIGH? RUN your program and see how it works when you answer YES and when you answer NO.

Let's use IF THEN to write a test for your friends.

Activity #2

First of all, we're going to have to tell your friends how to answer the test questions.
One easy way is to number each answer.
IF we number the questions THEN your friends don't have to type the answer - they only have to type the number of the answer.
That makes it easier for them.

Hang on .... Here we go.

```
 10 HOME
 20 PRINT "ANSWER THESE QUESTIONS BY PRESSING THE CORRECT NUMBER."
 30 PRINT:PRINT "WHAT IS ORANGE?"
```

Stop everything. Did you see what I did?
I put two PRINT statements on one line.
There is a colon between them. See that?
You can put more than one statement on a line - if you use a colon.
It is a handy way of letting you write a program without typing so many line numbers.
So line 30 has one blank PRINT statement and one PRINT statement.

```
 40 PRINT:PRINT "1.  RED AND YELLOW"
 50 PRINT "2. YELLOW AND BLUE"
 60 PRINT "3. RED AND BLUE"
 70 PRINT:INPUT A
 80 IF A=1 THEN 130
 90 PRINT "TRY AGAIN"
100 FOR T=1 to 1500:NEXT
110 HOME
120 GOTO 30
130 PRINT "GOOD"
140 END
```

I know you understood everything up to line 70.
Line 80 is that IF THEN choice. If the correct answer is given, the computer moves on to line 130 and skips everything else.
Line 90 is the reply to the incorrect answer.
Line 100 holds line 90 on the screen long enough for you to read it.
Line 110 clears the screen.
Line 120 says to go to the question - not the instructions. This program will go on and on until the correct answer is given.
Line 130 is the reply to the correct answer.

IF you can write one test question, THEN you can write several test questions.

Activity #3

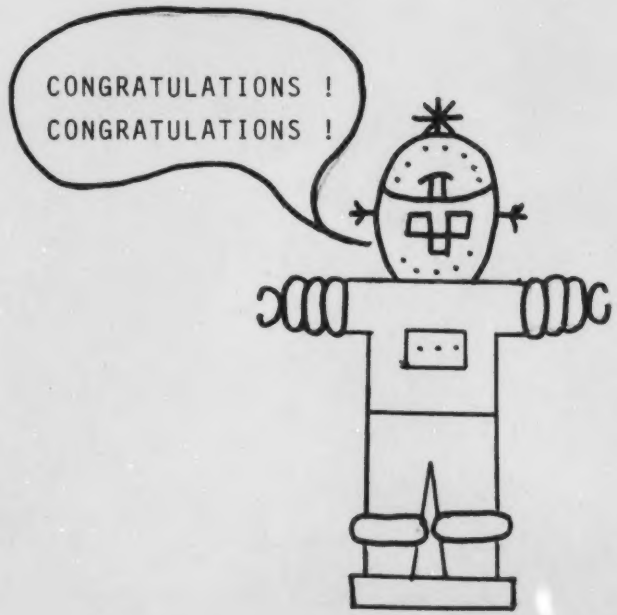
Use the same plan as Activity #2.

1. First clear your screen
2. Tell the person using your program how to answer the questions.
3. Print the question.
4. Print the possible answers.
5. Let them answer the question.
6. Tell the computer what to do if the answer is right.
7. Tell the computer what to print if the answer is wrong.
8. Tell the computer where to go to print the question again.
9. Now - instead of ending your program after the correct answer is given, type in another time delay routine so we have enough time to read the reply to the correct answer.
10. Type another question and keep going.

Sometimes this is too much typing to do all at once, especially if you are writing a question and answer test about E.T. or some other movie character.

You can write your program out on paper and ask a fast typist to type for you, or you can ask your teacher how to SAVE your program on disks or tapes - whatever system you have - and just do one question at a time. You can keep adding to your program as you have time.

(If you write a good Social Studies test, your teacher will probably help you find a typist!)



IF you have have accomplished this much, THEN you deserve to treat yourself.

Pat yourself on the back and say "WELL DONE".

TEST YOURSELF

1. The IF THEN statements allow you to write a program that includes a:
   1. choice
   2. print statement
   3. diet

2. IF cannot be used alone. You must also use:
   1. GOTO
   2. END
   3. THEN

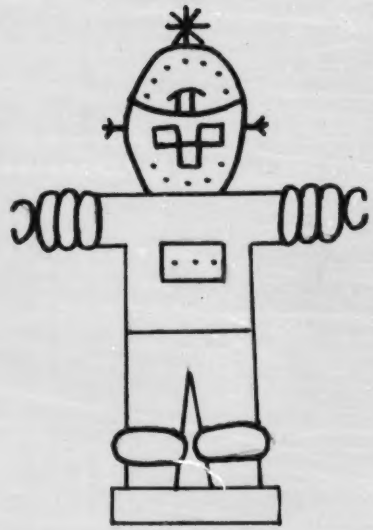
3. In this program, where is the computer instructed to go if the input is "NO"?

```
10?"CAN YOU COME OVER?"
20 INPUT A$
30 IF A$= "NO" THEN 60
40 ?"SEE YOU LATER."
50 END
60 ? "SORRY YOU CAN'T COME OVER."
70 END
```

WRITING MATH PROGRAMS

One kind of math program that can be written for the computer is one where the computer is asked to do the same kind of problem over and over.

```
 10 CLS

 20 ?"I KNOW THE NUMBER OF HOURS IN A DAY."

 30 ?"I WILL TELL YOU THE NUMBER OF HOURS THERE ARE"

 40 ? "IN AS MANY DAYS AS YOU WANT TO KNOW."
 45 ?
 50 ? "HOW MANY DAYS?"

 60 INPUT A

 70 B=24 * A

 80 ?"THERE ARE " B " HOURS IN " A " DAYS."

 90 ? "DO YOU WANT TO DO  THIS AGAIN?"

100 INPUT S

110 IF S= "YES" THEN 10

120 ? "GOOD BYE"

130 END
```



Line 60 is familiar to you.

Line 70 is the line that instructs the computer to multiply 24 (hours in a day) times A (the number that was typed in). B is the storage box that lets the computer do this problem over and over. It won't matter what A is. The computer will still be able to find the correct answer because of the information you put in it's B box. The information in the A box is changed each time you input new information.

```
 10 REM ADDING TWO NUMBERS AND SUBTRACTING ONE
 20 HOME
 30 ?"YOU AND YOUR FRIEND BUY SOME MARBLES."
 40 ?"BUT YOU LOSE SOME ON THE WAY HOME FROM THE STORE."
 50 ?:?:?:? "HOW MANY MARBLES DO YOU WANT TO BUY?"
 60 INPUT A
 70 ? "HOW MANY MARBLES DOES YOUR FRIEND WANT TO BUY?"
 80 INPUT B
 90 ?"HOW MANY DID YOU LOSE ON THE WAY HOME?"
100 INPUT C
110 ?"HOW MANY DO YOU HAVE LEFT?"
120 INPUT D
130 E = A + B - C
140 IF D = E THEN 170
150  ? "YOU LOSE.  THE ANSWER IS " E
160 END
170 ?"YOU'RE RIGHT! YOU WIN THE MARBLES."
180 END
```

We certainly used a lot of storage boxes; A,B,C,D, and E. I doubt if this program confused you at all. But just in case, here is an explanation of each line.

Line 10 was just to remind yourself what would be happening in the program.

Line 20 cleared the screen.

Line 30, 40 and 50 are PRINT statements. Did you notice that on line 50 several PRINT statements were on one line. And each statement was separated by a colon. You can put several statements on one line as long as you use the colon to instruct the computer that each statement is a separate one.

Lines 60, 80, 100 and 120 all asked for input. Each bit of information was put into a separate box for the computer to use later.

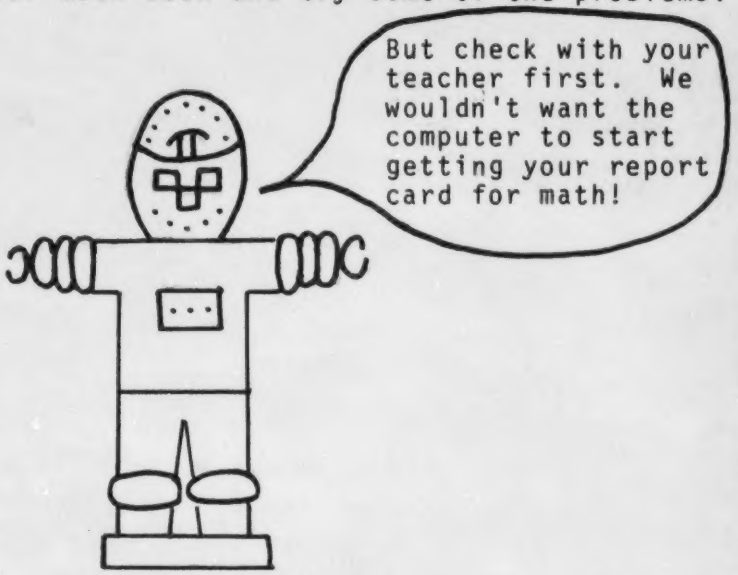
Line 130 told the computer what to do with the input it was receiving.

MEET YOUR COMPUTER, THE CALCULATOR

The computer will solve math problems for you in a flash. Try it. Ask the computer to do anything. You don't need a line number - just the PRINT statement.

```
? 3 + 4
? 405 + 873
? 2,034,539 - 53,467
```

Get your math book and try some of the problems.



The sign for multiplication is the * .
The sign for division is the / .
The computer seems to add, subtract, multiply, and divide almost at once, but it really reads your statement from the left side, then does the multiplication first, the division next, then the addition and last of all the subtraction.

```
? 6 + 4 * 7 / 2
```

In this statement, you can see that the computer does not work it the same way you would. There is a way to make the computer change it's ways. Put the problems you want the computer to do first inside parenthesis.

```
? (6 + 4) * 7 / 2
```

This time you should get the answer you expect.

The ↑ will even do math for you. 3 ↑ 5 means 3x3x3x3x3. If you recognize this from your math books as $3^{(5)}$ you know this is called exponentiation.

TEST YOURSELF

1. Writing a math computer program can allow you to:
   1. do the same kind of problem over and over
   2. waste your time
   3. INPUT A PRINT GOTO

2. E= A + B - C means:
   1. the computer will add letters
   2. the computer will add the value of A and B and subtract the value of C
   3. Education equals Ability and Brains minus Cans

3. If you tell the computer to PRINT 3 + 6, the computer will:
   1. print 9
   2. print 3 + 6
   3. print 36

4. You can use the computer just like a calculator
   1. True
   2. False

FOR NEXT The Automatic Counting Machine

Your computer has been programmed to recognize FOR and NEXT.
It knows something will be counted.

Imagine a race car track with someone counting how many times your car goes around the loop.
Three things happen:

1. Each time you pass the beginning, someone counts that loop around the track.
2. Each time you pass the beginning, someone waves you on.
3. When you have completed the required number of laps, they stop you.

That is what a FOR NEXT loop will do for your computer program.
A FOR NEXT statement begins with the word FOR and ends with the word NEXT and uses those storage boxes to keep track of how many times it is supposed to count.

Activity #1 - Time Delay Loops

```
10 FOR T= 1 to 500
20 NEXT T
```

Line 10 told the computer:

1. get ready to count
2. get storage box T ready
3. start counting at 1 and go to 500

Line 20 told the computer to go back and read line 10 again, and keep reading it one number at a time.

Since we didn't tell the computer to do anything except count, it counted silently until it reached 500. It took the computer 4 seconds to do this. (How fast can you count?)

You can write the FOR NEXT statement a little faster by putting it on one line.

```
10 FOR T= 1 TO 500:NEXT T
```

That is assuming you don't want the computer to do anything except count silently.

Let's move on and find out what else you can do with a FOR NEXT loop.

Activity #2

This time, let's tell the computer to PRINT the numbers from 1 to 500.
It doesn't go as fast as when it is counting silently - but it is still fast!

```
10 FOR T= 1 TO 500
20 PRINT T
30 NEXT T
```

Line 10 told the computer to put the numbers from 1 to 500 in the T storage box.
Line 20 told the computer to PRINT whatever was in the T storage box.
Line 30 said - go back and keep working.

This program is more fun to watch if we put that ; after the T in our PRINT statement.

```
20 PRINT T;
```

Activity #3

Let's make it a bit more complicated.

```
10 FOR A= 1 to 500    (get out your A storage box, computer)
20 PRINT A
30 FOR T= 1 to 50: NEXT T   (get another  storage box)
40 NEXT A
```

Did you get it?
Lines 10, 20, and 40 make up the loop that told the computer to count to 500 and print the numbers on the screen - just like the program in Activity #2, only we used the A storage box instead of T.

Line 30 was another loop. A plain old time loop that said slow down and count to 50 silently before you print the next number.

Activity #4

This time let's print your name all over the screen.

```
 1 HOME
10 FOR A= 1 to 123   (I got tired of typing 500)
20 PRINT "(type your name here and remember to leave a space) ";
30 FOR NEXT T= 1 to 50:NEXT T
40 NEXT A
```

Looks like those old GOTO programs - except this time the computer has been told to stop at number 123.

Activity #4

Let's practice that FOR NEXT loop.

Make the computer print:

COMPUTER PROGRAMMING IS FUN

10 times.

That was easy.

```
10 FOR A= 1 TO 10
20 PRINT " COMPUTER PROGRAMMING IS FUN "
30 NEXT A
```

Make this a bit more complicated, too.
Add :

```
line  1 HOME
line 21 FOR T= 1 TO 50  :  NEXT T
line 40 PRINT "AND A LITTLE BIT OF WORK."
```

Activity #5

More practice for you.

This time we will use 3 storage boxes.
Box A to tell the computer where to start counting.
Box B to tell the computer where to stop.
Box C to keep track of the numbers from the beginning to the end.

```
10 INPUT "WHERE SHALL I START COUNTING"; A
20 INPUT "WHERE SHALL I STOP COUNTING"; B
30 FOR C= A TO B
40 PRINT C;
50 NEXT C
```

Change line 40. Tell the computer to print a word or a sentence instead of just printing the numbers in the C box.

```
40 PRINT " I LIKE YOU ";
```

Maybe you would like to put a time delay loop after line 40. That really makes a complicated looking program. Go home and read to your mother and dad. They will think you are learning a new language - you are - BASIC!

TEST YOURSELF

1. FOR NEXT are:

   1. program statements
   2. words to tell the computer to count by 4's
   3. never heard of them

2. A FOR NEXT loop will:

   1. crash
   2. count for you
   3. print "See you next time!"

3. If you use the word FOR without the word NEXT, the computer will:

   1. burn out
   2. not notice
   3. print an error message

4, In this program how many times will IOWA be printed:

```
10 HOME
20 FOR A= 1 TO 10
30 PRINT IOWA
40 NEXT A
```

= is equal to	<= less than or equal to
< less than	>= greater than or equal to
> greater than	<> is not equal to

I hope you didn't think those math symbols were just in math books for fun! Computer programmers use them often to talk to their computers.

The "plain old English" meaning of these symbols is at the top of this page - just in case you forgot your math!

Here's a quick program to show you how to use these symbols.

Activity #1

```
10 IF 1 < 2 THEN PRINT "HI"
20 IF 1 > 2 THEN PRINT "BY"
```

RUN this. Which sentence will the computer print? Good. It will print line 10 because 1 is less than 2.

Activity #2

Change this program a bit with an INPUT statement.

```
10 INPUT A
20 INPUT B
30 IF A < B THEN ? "HI"
40 IF A > B THEN ? "BY"
```

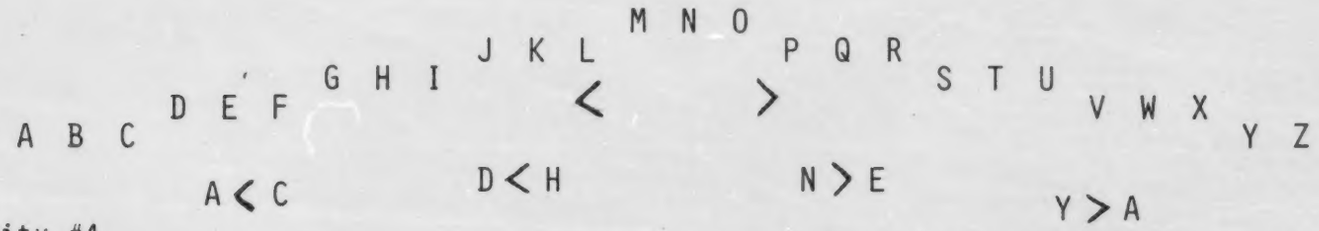
This time the words that are printed depend on which numbers you type.

Activity #3

This time we will use all of the statements given at the top of the page.

```
 1 HOME
10 PRINT "TYPE TWO NUMBERS."
20 PRINT "I WILL PRINT A MESSAGE FOR YOU"
30 INPUT A
40 INPUT B
50 IF A < B THEN PRINT "HI"
60 IF A > B THEN PRINT "HELLO"
70 IF A <= B THEN PRINT "WHAT A NICE DAY"
80 IF A >= B THEN PRINT "IT'S FUN TALKING TO YOU"
90 IF A <> B THEN PRINT "CAN YOU COME OUT AND PLAY"
100 IF A = B THEN PRINT "SEE YOU TOMORROW"
```

GREATER THAN AND LESS THAN APPLY TO ALPHABET LETTERS AS WELL AS TO NUMBERS



Activity #4

You can use < and > to put words in alphabetical order.
The computer knows the order of numbers and it knows alphabetical order too.

Being logical and orderly is what the computer is best at.

Try this:

```
 10 HOME
 20 PRINT "TYPE TWO WORDS FOR ME"
 30 PRINT "I WILL PUT THEM IN ALPHABETICAL ORDER FOR YOU."
 40 INPUT "WHAT IS YOUR FIRST WORD"; A$
 50 INPUT "WHAT IS YOUR SECOND WORD"; B$
 60 IF A$ > B$ THEN PRINT A$,B$
 70 IF A$ < B$ THEN PRINT B$,A$
 80 IF A$=B$ THEN PRINT "YOU TYPED THE SAME WORD TWICE"
 90 FOR T= 1 TO 2500: NEXT
100 INPUT "DO YOU WANT TO DO THIS AGAIN"; C$
110 IF C$= "YES" THEN GOTO 10
120 END
```

Line 60 said to the computer, "Look at the two words that were typed. If the first word that was typed begins with a letter closer to the beginning of the alphabet than the second word, then print the words in the order they were typed. If the second word that was typed begins with a letter close to the beginning of the alphabet than the first word, then print the words in reverse order."

Here's a program to practice your <> skills.

Activity #3

We will ask the person using our program to tell us the time.

The statements that we have the computer print will depend on their answer.

```
10 HOME
20 INPUT "TELL ME WHAT TIME IT IS THIS MORNING";A
30 PRINT A "O'CLOCK"
40 IF A= 12 THEN GOTO 90
50 IF A<8 THEN PRINT "IT'S A GOOD TIME FOR SLEEPING"
60 IF A>8 THEN PRINT "IT'S TIME FOR WORKING
70 IF A= 8 THEN PRINT "TIME TO GET UP"
80 END
90 PRINT "MIDNIGHT.  ALL'S WELL"
```

In line 40 we sent the computer directly to line 90. We had to. Do you see why?

In line 50 any number that was smaller than 8 was given a response that showed they were sleeping.

In line 60 any number that was bigger than 8 was given a response that showed they were working.

Is 12 bigger than 8? It sure is. Is 12 o'clock a.m. a time for working? No. So it had to have it's own response.

It also had to be somewhere in the program where the computer would not read the response in line 60 and print it.

If you don't see why this had to be done, try moving line 90 to line 40 and see what happens.

These<> and = signs can help you when you want the computer to compare numbers or letters.

TEST YOURSELF

1. < > are used in computer programs to:

   1. confuse people
   3. to decorate the screen
   3. to compare the value of numbers or letters

2. In this program what will the computer print?

```
10 IF A< F THEN PRINT "TRUE"
20 IF A> F THEN PRINT "FALSE"
```

   1. True
   2. False
   3. both True and False

3. What will the computer print?

```
10 A= 10: B= 20
20 IF A< B THEN GOTO 40
30 IF B> G THEN GOTO 50
40 PRINT "A< B"
50 PRINT "A> B"
```

   1. A< B
   2. A> B
   3. I HAVE A HEADACHE

4. You would use the IF THEN statements and < > to:

   1. instruct the computer to make a choice, then do something
   2. to make your program more interesting
   3. print the math problems that are in the back of the book

When you fill a hat with slips of papers with numbers on them and pick a number without looking at the slips, you are picking a number at random.

Your computer can do this for you.

All you have to do is tell that computer that is what you want it to do. And tell it in BASIC.

To make the Apple computer print random numbers that we can use in our programs, we have to give it a rather long command.

```
INT(10*RND (1)) + 1
```

INT - tells the computer to print whole numbers - we don't want decimals. INT stands for integer, which is a $2.00 word for whole numbers.

(10 tells the computer the biggest number we want it to pick from - in other words, "Computer, pick any number from the beginning to the number 10". Of course, you can change this number depending on what you want from the computer.

*RND(1) tells the computer to pick numbers in random order. Don't worry about the * and the (1) for now. They just have to be there. Someday when you are a famous computer scientist you will need to know. But not yet!

+1 tells the computer to begin at the number 1. You can change this number, depending on what you want from the computer.

Here's the way it looks again:

```
INT(10*RND (1)) + 1
```

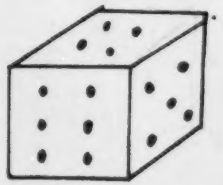
Pay special attention to the parenthesis. We have to use two complete sets, one set inside the other. Don't leave anything out!

Activity #1

This time we will write a program for the computer to toss the dice in a game for you.

Each die has 6 numbers. You want the computer to toss both die.

```
 1 HOME
10 FOR D= 1 to 2
20 INT(6 * RND(1)) + 1
30 NEXT D
```



Line 10 told the computer to toss the die 2 times.

Line 20 told the computer to pick a number from 1 to 6 and print it.

You can play a game. Program the computer to pick numbers from 1 to 1000 RND (1000). You take the first number that is printed - your partner the second. See who gets the bigger number most often.

Activity #2

We can put a letter in parenthesis after RND instead of a number. That will let the computer know we are using it's storage boxes. Of course, we will have to tell the computer what to put in it's storage boxes.

Let's try this:

```
10 HOME
20 PRINT "I WILL PRINT RANDOM NUMBERS FOR YOU"
30 INPUT "WHAT IS THE BIGGEST NUMBER I SHOULD USE" ; A
40 INPUT "HOW MANY NUMBERS SHOULD I PRINT" ; B
50 FOR N= 1 TO B
60 PRINT INT (A * RND (1)) + 1
70 NEXT N
```

Reading this program should be a snap for you by now.

line 10 - perfectly clear, I hope
line 20 - simple PRINT statement
line 30 INPUT the largest random number you want (put in A storage box)
line 40 INPUT the number of times the computer should print a random number (put in B storage box)
line 50 use the N storage box to count the number of times the computer is printing a number
line 60 - simple print statement
line 70 - part of the FOR NEXT counting loop

Activity #3

Let's play I'VE GOT A SECRET NUMBER.

```
10 HOME
20 PRINT "TRY TO GUESS  MY NUMBER?"
30 PRINT "THE NUMBER IS BETWEEN 1 AND 5"
40 N= INT (5 * RND (1)) + 1
50 INPUT "YOUR GUESS": G
60 IF G <>N THEN PRINT "SORRY"
70 IF G= N THEN PRINT "YOU GOT IT"
80 FOR T= 1 TO 1000: NEXT
```

Would you like to see the computer's number?

Then put in a line between 70 and 80 and tell the computer to PRINT the random number it had picked.

```
75 PRINT "MY NUMBER WAS": N
```

I hope you understood that the computer put it's random number in it's N box and put your guess in the G box.

Then it compared the N and the B boxes to see if they were the same or different.

TEST YOURSELF

1. RND stands for:
   1. round
   2. random
   3. Real Nice Day

2. Print RND (25) means:
   1. the computer will print 25 all over the screen
   2. there are 25 lines in your program
   3. he computer will print numbers from the beginning to 25 in a random order

3.
```
10 A= INT (25 * RND (1)) + 1
20 B= INT (25 * RND (1)) + 1
30 C= A+B
40 PRINT A "+" B "="
50 INPUT D
60 IF D= C THEN PRINT "OK"
```

   In which line of the program will the computer actually solve the math problem:
   1. line 10
   2. line 20
   3. line 30

4.
```
10 A= INT (9 * RND (1)) + 1
20 B= INT (100 * RND (1)) + 1
30 C= B*A
40 PRINT B"*" A "=" C
```

   How large will the largest number be in A?
   1. 9
   2. 100
   3. can't tell

5. How large will the largest number be in B?
   1. 9
   2. 100
   3. can't tell

You've been using variables all along.
I didn't call them that because I was sure the word variable would scare you off.

Variables are those letters we use to tell the computer to use it's storage boxes.

We call those letters variables because we can change (or vary) their use.

Activity #1

Here's a program that will add a list of numbers for you.

When you sit down to add numbers, first you get the list of numbers.
Then you add them together.

That is what this program will do.
If you were a computer following this program, you would get the list and put it in a storage box labeled L(list of numbers).

Then you will add the list and put the totals in a box labeled A (for add).

This is how you tell the computer to do that in BASIC language.

```
 10 HOME
 20 ?"I WILL  ADD NUMBERS FOR YOU."
 30 ?"?"WHEN I PRINT A ? YOU ENTER A NUMBER."
 40 ?"WHEN YOU WANT ME TO STOP TYPE ØØ
 50 ?"I WILL PRINT THE SUM"
 60 A=0
 70 INPUT X
 80 IF A= 00 THEN 110
 90 A=A+X
100 GOTO 70
110 ?:?:?"THE SUM IS " ; A
```

Line 60 said to get box A ready and no matter what - start with 0.
Line 70 said, get the numbers and put them in Box X.
Line 80 said, if they signal they are done, go to 110.
Line 90 said add the numbers in X to the numbers in A and put the sum in A.
Line 100 said to back and get another number to add.
Line 110 said to print the answer on the screen.

Activity #2

Remember the I've Got a Secret Number program?

We're going to add some lines that will keep track of how many guesses you made.

And we'll make it a bit more interesting by telling the computer to give us clues to tell us if the number we guess is too big or too small.

```
 10 HOME
 20 PRINT "I WILL THINK OF A NUMBER FROM 1 TO 100."
 30 PRINT "GUESS MY NUMBER"
 40 X= INT (100 * RND (1)) + 1
 50 PRINT: INPUT "YOUR GUESS IS";G
 60 IF G > X THEN PRINT "TRY A BIGGER NUMBER": GOTO 50
 70 IF G < X THEN PRINT "TRY A SMALL NUMBER": GOTO 50
 80 IF G=X THEN PRINT "THAT'S IT"
 90 PRINT "YOU GOT  IT IN" A " GUESSES."
100 FOR T= 1 TO 2000: NEXT
 45 A=0
 55 A+A+1
```

In line 40 we set the range of numbers for the computer to pick from at 100.

Line 45 set the counter box at 0 (it was placed before the INPUT statement).

Line 55 said to add the number of guesses made (it was placed right after the INPUT G statement).

Lines 60, 70 and 80 told the computer what to print after the guess was made.

Then we told the computer to check the guess with the number it had picked and to print the statement that fit.

Activity #3

I hope you are ready for a mind-boggler.
I think you should be able to understand this program without too much trouble.

This program will print math problems.
Not only will it print them, but it will count the number of problems that are done correctly.

Included in this program are several REM statements.
This is new to you.
REM stands for REMARK. REM statements stay in your LIST of instructions, but the computer does not read them or do anything about them. REM statements are for people, not for computers!

You can type your program with or without the REM statements. They make no difference in running a program.

```
 10 HOME
 20 REM SET THE COUNTER FOR THE CORRECT ANSWERS
 30 A=0
 40 PRINT "I WILL PRINT TEN NUMBER PROBLEMS FOR YOU"
 50 PRINT "THEY WILL BE ADDITION PROBLEMS."
 60 PRINT "PLEASE TELL ME THE LARGEST NUMBER I CAN USE."
 70 INPUT R
 80 PRINT "THANK YOU.  I WILL NOT USE ANY NUMBERS LARGER THAN"; R
 90 PRINT
100 REM SET THE LOOP FOR COUNTING 10 PROBLEMS
110 FOR D= 1 TO 10
120 REM COMPUTER PICKS NUMBERS AT RANDOM AND PRINTS ON THE SCREEN
130 X= INT (R * RND (1)) + 1
140 Y= INT (R * RND (1)) + 1
150 PRINT X "+Y"=";
160 REM COMPUTER SOLVES PROBLEMS AND PUTS ANSWER IN IT'S Z BOX
170 Z= X + Y
180 REM STUDENT TYPES ANSWER
190 INPUT C
200 REM COMPUTER CHECKS STUDENTS ANSWER
210 IF C <> Z THEN PRINT "NO: GOTO 260
220 IF C= Z THEN PRINT "YES"
230 REM CORRECT ANSWER IS COUNTED
240 A= A+1
250 REM END OF LOOP FOR COUNTING 10 PROBLEMS
260 NEXT D
270 PRINT "YOU GAVE" A" CORRECT ANSWERS"
```

TEST YOURSELF

Use the program in Activity 3 to answer these questions.

1. What will A do for you?

   1. count the correct answers
   2. count all the problems
   3. set space aside for the problems

2. What will R do?

   1. come after Q
   2. hold the number that will be the largest number the computer can use
   3. count the problems

3. What will D do?

   1. print numbers from 1 to 10 on the screen
   2. equal R
   3. count the number of problems

4. What will X and Y do?

   1. hold the random number picked by the computer
   2. confuse everything
   3. add

5. What will Z do?

   1. end the program
   2. nothing - it isn't necessary
   3. hold the answer to X+Y

6. What will C do?

   1. hold the student's answer
   2. add X+Y
   3. count the student's answer

7. What is REM for?

   1. to stop the program
   2. the explanation of how the program works
   3. counts all blotches

Lesson 10

PUTTING BASIC TO USE IN MATH

If you've been learning the multiplication tables, you'll have fun with this one.

Activity #1

This program will print multiples of 2 up to 2*10.

```
10 HOME
20 A= 0
30 FOR N= 1 TO 10
40 A= 2*N
50 PRINT A;
60 NEXT N
```

(sets the counter for 10 problems)

Could we change line 50 to print the problem as well as the answer?

```
50 PRINT "2 * " N "=" N*2
```

In regular English we said, "Computer, print a number 2 and the multiplication sign.
Then print whatever is in your N box.
Then print an = sign.
Then multiply whatever is in your N box by 2 and print that."

Activity #1A

We can write this same program another way.

It's time you learned about STEP. Not only do the FOR NEXT loops count for you, they will count by 2's, by 3's, by anything you tell them to, as long as you tell them in BASIC. It is like saying, "Go up the steps 2 at a time, or 3 at a time, or 4 at a time."

```
10 HOME
20 FOR A=0 TO 50 STEP 5
30 PRINT " 5 * " A "=" 5*A
40 NEXT A
```

Activity 1B

This time you write a program that will add 2 to every number from 1 to 100.

Activity #2

You know the number of hours in a day and the number of minutes in an hour.

Let's use that to write a simple computer program that will do all the math for you.

Type instructions to the people using your program.

You put in the line numbers.

JUST IMAGINE

Suppose you are on a bicycle trip across the United States. You have been riding 10 days non-stop. You would like to know how many hours you have been riding. You would also like to know how many minutes.

Now for the program.

You will need a storage box to keep track of the number of days (D), and a storage box to keep track of the number of hours (H), and another storage box to keep track of the number of minutes (M).

Add a box to count from 1 to 10 (C).

```
100 CLS
110 PRINT "DAYS", "HOURS", MINUTES"
120 D= 1: H= D * 24: M= H * 60
130 FOR C= 1 TO 10
140 PRINT D*C, H*C, M*C
150 NEXT C
```

Activity #3

You can write simple programs for any math situation where you know what to and how to do it.

Can you figure the area of a room? Do you know how to di it? Then you can write a computer program.

Do you know how many pounds are in a to ? Do you know how to find out the number of pounds in 75 tons?

Do you know the cost of 1 dozen rolls at 13¢ each? How do you find the cost of 5 dozen or 25 dozen?

Do you know how many quarts in a gallon? How many pints? How many cups?

Let's take the last one and write it out.

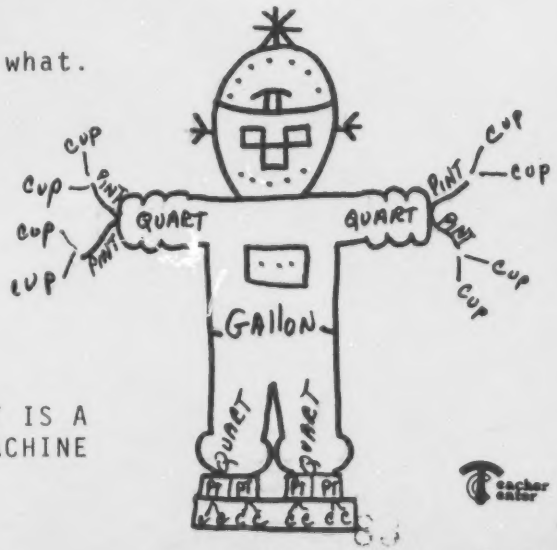
Activity #3A

First, let's set up our storage boxes.
G for the number of gallons.
Q for the number of quarts in gallons (that would be 4*G) - 4 quarts in a gallon.
P for the number of pints in a gallon (that would be 2*Q) - 2 pints in a quart.
C for the number of cups in a gallon (that would be 2*P) - 2 cups in a pint.

Add a counting loop to count the number of times we want this information printed. K

Add a heading statement so we don't forget which number tells us what.

```
10 HOME
20 ?"GALLONS", "QUARTS", "PINTS"
25 G=1: Q=4: P=2
30 FOR K= 1 TO 12
40 PRINT G, G*Q, G*P
50 NEXT K
```



Let's print out something hard to figure.

Change line 30 to:

```
30 FOR K= 50 TO 500 STEP 45
```

REM ROBOT IS A
MEASURING MACHINE

TEST YOURSELF

1. In a FOR NEXT loop STEP 6 tells the computer to:

   1. count by ones six times
   2. count by sixes
   3. print 6 steps on the screen

```
10 HOME
20 PRINT "THIS PROGRAM WILL COMPUTE THE"
30 PRINT "AREA OF A RECTANGLE"
40 INPUT "WHAT IS THE LENGTH IN FEET"; L
50 INPUT "WHAT IS THE WIDTH IN FEET"; W
60 A= L*W
70 PRINT "LENGTH; "WIDTH, "AREA"
80 PRINT L,W,A
```

2. In this program, which line tells the computer to do the math problem?

   1. 40
   2. 50
   3. 60

3. Which lines give the information to the computer so the computer can solve the problem?

   1. 20 and 30
   2. 40 and 50
   3. 60 and 80

GOSUB

This is a very quick lesson on GOSUB. I want you to know about it, especially if you are going to be writing programs using graphics. A subroutine is a good place to write a graphics program.

Very simply, a subroutine is usually something you want the computer to do several times in your program. Instead of typing it over and over, you just put the statements somewhere out of the way and tell the computer to go get it and do what it says at the proper time.

```
 10 HOME
 20 INPUT "TELL ME A NUMBER YOU WANT TO ADD";A
 30 INPUT "TELL ME ANOTHER NUMBER YOU WANT TO ADD";B
 50 INPUT A "+" B "="; C
 60 D = A +B
 70 IF D = C  THEN GOSUB 500
 80 GOTO 10
500 PRINT "##################"
510 PRINT "##################"
520 PRINT "   YOU'RE RIGHT   "
530 PRINT "##################"
540 PRINT "##################"
545 FOR T= 1 TO 1000: NEXT
550 RETURN
```

This program could be made a lot better. There is no way to end it except with the BREAK key or Cntl C.

Did you notice that there was a new word. RETURN. After the computer reads a GOSUB statement, it returns to the place in the program where it was reading. If it was reading line 70, it went back to the next line (80) and continued on it's merry way.

You can make this easier to use by adding a FOR NEXT loop to set limits on the number of times this program runs.

```
15 FOR C= 1 TO 10
85 NEXT C
```

And to end the program - put a line before line 85.

```
83 IF C= 10 THEN END
```


Teacher Center

STRINGING ALONG

Or Concantination (if you like big words)

Concantination is the process of adding one string to another.

Your computer can make up silly sentences for you.

If you put parts of sentences in the computer's storage boxes and then add those parts together, you will have sentences. You can add the parts together in different ways. Or use INPUT to change some of the parts of the sentences.

Let's take a look at one program that can be funny to RUN.

Activity #1

```
 1 HOME
10 ?"NOUN REVIEW"
20 ?:?:INPUT "TYPE THE NAME OF A PERSON"; A$
30 INPUT "TYPE THE NAME OF A PLACE";B$
40 INPUT "TYPE THE NAME OF A THING"; C$
50 ?:? A$ "WENT TO" B$ " IN A "C$
```

Run this one.

You may have to tell your computer what you are doing so it will save space for you to type sentences. One way to do this is to type a line number 5 and a statement that says - save space computer.

```
5 CLEAR (100)
```

Now I've got space saved for 100 letters. That ought to be enough. You may not need this statement. Apple user's CLEAR will erase whatever is in your string - not save space. It won't hurt this program, but it might if you are not aware of the difference in the meaning of CLEAR.

Activity #2

Here's a program that lets you change the verb.

```
10 HOME
20 CLEAR (100)
30 A$= "MRS. GOODWELL"
40 INPUT "TYPE A VERB - PAST TENSE" B$
50  C$= "ON A CHAIR"
60 ? A$ + B$ + C$
```

Before you let others use this program you will want to type in some instructions so they know what to do when the question mark appears on the screen. You can add a statement so the program will go back to the beginning instead of stopping at line 60.

Activity #3

Here is one variation the kids in your class will like.

```
10 HOME
20 CLEAR 100
30 INPUT "TYPE AN ADJECTIVE"; A$
40 INPUT " TYPE A VERB - PAST TENSE"; B$
50 PRINT "(your teacher's  name)" + B$ + A$ + "(type in your grade) GRADE CLASS."
```

Activity #4

```
10 HOME
20 CLEAR 100
30 PRINT "HERE IS A SENTENCE"
40 PRINT "TYPE AN ADJECTIVE THAT MAKES SENSE."
50 PRINT " THE "        " BIRD IS FLYING."
60 INPUT A$
70 PRINT " THE "  A$ " BIRD IS FLYING."
```

Activity #5

Type in your own program.

Ask the computer user to type in a noun.
Can you do that?

TEST YOURSELF

1. GOSUB

   1. is a new kind of sandwich
   2. tells the computer to look for another line in your program
   3. is an underwater village

2.
```
1000 PRINT "HI"
1010 FOR T= 1 TO 500: NEXT T
1020 RETURN
```

   The word that let's you know this as a subroutine is
   1. FOR
   2. NEXT
   3. RETURN

3. When a computer finishes a subroutine it RETURNS:
   1. to the beginning of the program
   2. to look for an END statement
   3. to continue from where it was in your program

4. Strings can be:
   1. added to each other
   2. changed to money
   3. rolled into a ball

5. You can learn a lot more about strings by:
   1. using your computer manual
   2. watching the clouds float by
   3. reading a book called "How to Macrame"

Lesson 12

READ DATA

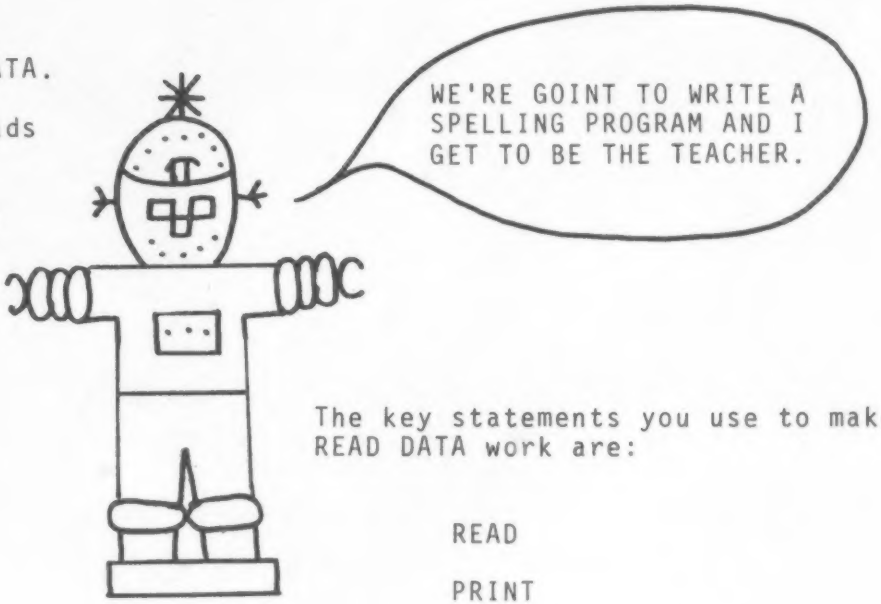
When the computer sees the word READ it looks for a program statement called DATA.

When it finds the DATA statement, it reads the words in the DATA statement.

DATA statements contain lists of things.

This is how the computer has been programmed to operate:

1. The computer sees the word READ.
2. The computer looks for a DATA statement. The DATA statement can be anywhere in your program
3. The computer reads the first word in the DATA list.
4. The computer puts the word in a storage box. The READ statement told it which box.
5. The computer PRINTS the word.
6. The computer READS the next word in the list and does the same thing with it. The computer uses a counter - like you would use your finger to keep your place in a book. The counter lets the computer keep track of which words it has used.



The key statements you use to make READ DATA work are:

READ

PRINT

DATA

Now that you know how READ DATA work, let's move on and find out why you would use these statements in a program.

Spelling Program Using READ DATA

```
10 HOME
20 DATA "COW", "HORSE", "MULE", "DOG"
30 DATA "CAT", "MOUSE", "FISH", MONKEY"
```

DATA statements can be placed anywhere in your program. These are at the beginning but the computer doesn't care. You could put them in line 1000 and the computer would still find them.

The one thing that is important is punctuation. That cannot be changed.

You can change the words in the spelling list by typing new DATA statements. That is all you will have to shange.

```
40 READ A$ (C)
```

This tells the computer to READ the DATA list. I'm sure you noticed the extra letter in parenthesis. Each of those storage boxes has little boxes inside. I want to use the A$ box and in the A$ box I want the computer to put the words from the DATA list in the C box. Later on in this program, I will ask the computer to print all the words in the DATA statements and the INPUT statements. They had to be put in a box inside the A$ box or the computer would have pushed them out everytime a new word was put in the A$ box.

```
50 PRINT A$ (C)
```

The computer will print the words in the DATA list.

```
60 INPUT Z$ (C)
```

You will type the spelling word the computer has just printed on the screen. The computer will put your spelling it it's Z$ (C) box.

```
70 IF A$ (C)= A$(C) THEN PRINT "GOOD"
80 IF A$ (C)=<>Z$(C) THEN PRINT "TOO BAD"
```

We're checking to see if the word was spelled correctly. Did the input= the spelling in the DATA statement?

Stop everything and RUN this program. As you can see, we have a few more things to do before we can use this program.

We need a few time delay loops! We need the spelling word held on the screen for a few seconds, then a clear screen. We need our messages for correct and incorrect spelling held on the screen long enough to read them.

```
81 FOR T= 1 TO 1500: NEXT
51 FOR T= 1 to 2000: NEXT
52 HOME
```

Run your program again. Are the words flashed on the screen, then off? Do the messages stay there long enough to be read?

Now, let's get all the DATA READ by the computer. We'll do that with a counting loop.

```
11 FOR J= 1 TO 8
90 NEXT J
```

There are 8 words on your list. We want the computer to read all of them.

Would you like the spelling words numbered?

```
15 PRINT "PROBLEM#" J
```

Let's keep track of the number of right answers and the number of wrong answers.

```
75 IF A$(C)= Z$(C) THEN R=R+1
85 IF A$(C)<>Z$(C) THEN W=W+1
```

Now to tell the computer what to do if it reaches the end of the program. The variable J is counting the number of problems.

```
 89 IF J= 8 THEN 100
100 PRINT "RIGHT", "WRONG"
```

The comma will do the spacing for us on the TV screen.

The computer is reading all eight words in the spelling list and printing both the DATA list and the INPUT.

```
110 FOR J= 1 TO 8
120 PRINT:PRINT A$(C), Z$(C)
130 NEXT J
140 END
```

This program can be made even better with a few more statements that would place the words on the screen differently. And you know how to make the program take more than 8 words. And change the time delays.

When you LIST this program, it does look like a foreign language. But I hope by now you are very comfortable reading BASIC.

TEST YOURSELF

```
10 HOME
20 PRINT "I WILL PRINT WORDS FOR YOU."
30 FOR C= 1 TO 3
40 READ A$
50 PRINT A$
60 FOR T= 1 TO 200: NEXT T
70 NEXT C
80 DATA "COME", "HERE", "JACK"
```

1. In this program line 30 says:

   1. Read and do lines 30, 40, 50, 60, and 70 three times.
   2. PRINT 1, 2, 3 on the TV screen.
   3. Go to the car three times.

2. Line 40 tells the computer to:

   1. Look for a DATA statement and read it one word at a time.
   2. Print one word from the DATA statement.
   3. Read a book on computer programming.

3. Line 50 tells the computer to:

   1. print READ
   2. print the next word in the DATA statement
   3. print nothing at all

```
10 HOME
20 PRINT "I WILL PRINT WORDS FOR YOU."
30 READ A$
40 FOR C= 1 TO 3
50 PRINT A$
60 FOR T= 1 TO 200:NEXT T
70 NEXT C
80 DATA "COME", "HERE", "JACK"
```

4. In this program, the counting loop is after the READ statement. The PRINT statement is in the FOR NEXT loop.

   1. COME
HERE
JACK
   2. COME
COME
COME
   3. An error message will appear on the screen.

ANSWERS TO TEST YOURSELF

Introduction, p. 4

1. 2
2. 1
3. 2
4. 3
5. 2
6. 2
7. 3

Lesson 1, p. 8

1. 1
2. 1
3. 2
4. 3
5. 1
6. 1
7. 1
8. 2
9. 2
10. 3
11. 2
12. 3

Lesson 2, p.12

1. 1
2. 2
3. 1
4. 2

Lesson 3, p. 16

1. 3
2. 1
3. 2
4. 1
5. 1
6. 2

Lesson 4, p.20

1. 1
2. 3
3. line 60

Lesson 5, p. 24

1. 1
2. 2
3. 1
4. 1

Lesson 6, p. 28

1. 1
2. 2
3. 3
4. 10

Lesson 7, p. 32

1. 3
2. 1
3. 1
4. 1

Lesson 8, p. 33

1. 2
2. 3
3. 3
4. 1
5. 2

Lesson 9, p. 40

1. 1
2. 2
3. 3
4. 1
5. 3
6. 1
7. 2

Lesson 10, p.44

1. 2
2. 3
3. 2

Lesson 11, p.48

1. 2
2. 3
3. 3
4. 1
5. 1

Lesson 12, p. 52

1. 1
2. 2
3. 2
4. 2

COMPUTER PROGRAMMING CONTRACT

Part One

I, _____, agree to read and practice the following computer lessons:
(student)

Lessons # _____

What hours will time be available on the computer? _____

When will conferences be held with the teacher? _____

When will you take the tests at the end of each unit? _____

Part Two

_____ agrees to write one computer program for the class to use.

When will you write your program? _____

When will you meet with your teacher to discuss your program? _____

When will you complete your program? _____

How will you share this program with others? (As part of a class lesson?) _____

_____ Date

_____ (Signature)
(student)

_____
(teacher)

STUDENT RECORD SHEET

COMPUTER PROGRAM COMPLETED

STUDENT NAME